

AMENDMENTS TO THE CLAIMS

1 1. (Canceled).

1 2. (Currently Amended) A method for managing changes in a computer system, the

2 method comprising the steps of:

3 storing, in a storage space, undo information for removing changes that are being

4 made to data by a plurality of entities, wherein application of the undo

5 information returns the data to which the changes were made to a state of the

6 data that existed prior to the changes, wherein the undo information for each

7 entity of the plurality of entities is stored in a segment of a plurality of

8 segments within said storage space;

9 monitoring usage of the storage space by the entities; wherein said step of storing

10 undo information includes at least assigning each entity with a segment for

11 storing the undo information of the entity based on the usage; and

12 automatically adjusting ~~the particular amount of storage that is dedicated to the~~

13 ~~circular buffer by adjusting~~ at least one of the number of segments in the

14 plurality of segments, and the sizes of the plurality of segments based on the

15 usage.

1 3. (Previously Presented) The method of Claim 54, further comprising the steps of:

2 establishing a maximum amount of the storage space; and

3 preventing a sum of the sizes of the plurality of segments from exceeding the

4 maximum amount of the storage space.

1 4. (Currently Amended) A method for managing changes in a computer system, the
2 method comprising the steps of:
3 storing, in a storage space, undo information for removing changes that are being
4 made to data by a plurality of entities, wherein application of the undo
5 information returns the data to which the changes were made to a state of the
6 data that existed prior to the changes, wherein the undo information for each
7 entity of the plurality of entities is stored in a segment of a plurality of
8 segments within said storage space;
9 monitoring usage of the storage space by the entities; and
10 automatically adjusting at least one of the number of segments in the plurality of
11 segments and the sizes of the plurality of segments based on the usage;
12 wherein said step of monitoring usage further includes at least the step of monitoring
13 usage in each period of time for a series of periods of time, and the step of
14 automatically adjusting is based at least on a comparison between a usage of a
15 first period of time and a usage of a second period of time.

1 5. (Original) The method of Claim 4, said step of automatically adjusting the plurality of
2 segments further comprising the steps of:
3 determining whether usage has decreased over a predetermined time based at least in
4 part on the usage in one or more periods of time of the series of periods of
5 time; and
6 if usage has decreased over the predetermined time, then shrinking a sum of the sizes
7 of the plurality of segments.

6. (Original) The method of Claim 5, said shrinking including deleting a segment from the plurality of segments.

7. (Original) The method of Claim 4, said step of automatically adjusting the plurality of segments further comprising the steps of:

determining whether usage has decreased over a predetermined time based at least in part on the usage in one or more periods of time of the series of periods of time; and

if usage has decreased over the predetermined time, then

determining whether a first amount of storage space allocated to a first

segment of the plurality of segments is being used by a first entity

storing undo information in the first segment, and

if the first amount is not being used by the first entity, then de-allocating the

first amount from the first segment.

8. (Original) The method of Claim 7, said step of automatically adjusting the plurality of segments further comprising the steps of:

determining whether de-allocating the first amount leaves an amount allocated to the first segment that is less than a predetermined minimum amount; and

if de-allocating the first amount leaves less than the predetermined minimum amount, then deleting the first segment.

9. (Canceled).

1 10. (Previously Presented) The method of Claim 54, said step of automatically adjusting
2 the plurality of segments further comprising the steps of:

3 determining whether first conditions are satisfied; and
4 if the first conditions are satisfied, forming a new segment to store undo information
5 for a new entity; wherein the first conditions include that at least a first
6 amount of the storage space is not allocated to any segment of the plurality of
7 segments.

1 11. (Previously Presented) The method of Claim 54, said step of automatically adjusting
2 the plurality of segments further comprising the steps of:

3 determining whether first conditions are satisfied; and
4 if the first conditions are satisfied, forming a new segment to store undo information
5 for a new entity, wherein the first conditions include that every segment of the
6 plurality of segments stores undo information for at least one entity of the
7 plurality of entities.

1 12. (Previously Presented) The method of Claim 2, said step of storing undo information
2 further comprising the steps of:

3 determining whether a first segment of the plurality of segments is not storing undo
4 information for the plurality of entities; and
5 if the first segment is not storing the undo information for the plurality of entities,
6 storing undo information for a new entity in the first segment.

1 13. (Previously Presented) The method of Claim 2, said step of automatically adjusting
2 further comprising:

determining based on the usage whether a first amount of the storage space allocated to a first segment is not currently used by the plurality of entities; and if the first amount is not currently used, associating a new entity with the first amount of the storage space.

14. (Previously Presented) The method of Claim 54, wherein:
a first entity of a plurality of entities is associated with a first segment of the plurality of segments; and
said step of automatically adjusting the plurality of segments further comprises increasing a size of the first segment in response to a request from the first entity by allocating an additional amount of the storage space to the first segment.

15. (Original) The method of Claim 14, said step of automatically adjusting the plurality of segments further comprising:
determining whether sufficient storage space is already allocated to the first segment for storing undo information included in the request from the first entity; and
if it is determined that sufficient storage space is not already allocated to the first segment, then performing said step of increasing the size of the first segment.

16. (Original) The method of Claim 15, wherein a sum of the additional amount and the storage space already allocated to the first segment is sufficient for storing the undo information included in the request from the first entity.

17. (Original) The method of Claim 15, wherein the additional amount is based on the storage space already allocated to the first segment.

1 18. (Original) The method of Claim 14, wherein the additional amount is selected from a
2 plurality of predetermined amounts.

1 19. (Original) The method of Claim 14, said step of allocating the additional amount
2 further comprising:

3 determining whether the additional amount of the storage space is available in storage
4 space not currently allocated to the plurality of segments; and
5 if the additional amount of the storage space is available in storage space not
6 currently allocated to the plurality of segments, then obtaining the additional
7 amount of storage space from the storage space not currently allocated.

1 20. (Original) The method of Claim 14, wherein:

2 a set of one or more entities of the plurality of entities is alone associated with a
3 second segment of the plurality of segments; and
4 said step of allocating the additional amount further comprises:

5 determining whether the additional amount of the storage space is currently
6 allocated to the second segment of the plurality of segments and is not
7 used by the set of one or more entities; and

8 if the additional amount of the storage space is currently allocated to the
9 second segment and is not used by the set, then obtaining the
10 additional amount of storage space by de-allocating from the second
11 segment the storage space currently allocated to the second segment
12 and not used by the set.

21. (Original) The method of Claim 4 said step of monitoring usage in each period of time further comprising the step of monitoring an amount of the undo information stored in each period of time.

22. (Original) The method of Claim 4, said step of monitoring usage in each period of time further comprising the step of monitoring a number of entities started in each period of time.

23. (Original) The method of Claim 4, said step of monitoring usage in each period of time further comprising the step of monitoring a maximum number of entities executing concurrently in each period of time.

24. (Original) The method of Claim 4, said step of monitoring usage in each period of time further comprising the step of monitoring a maximum duration in each period of time among durations of queries terminating during the period of time, said queries using at least some of the undo information stored in the storage space.

25. (Original) The method of Claim 14, wherein the additional amount is an extent of contiguous storage space.

26. (Previously Presented) The method of Claim 54, the step of automatically adjusting the plurality of segments further comprising the steps of:

allocating unused amounts of the storage space to the plurality of segments in response to receiving undo information from the plurality of entities; and de-allocating unused amounts of the storage space from the plurality of segments periodically.

1 27. (Original) A method for managing changes in a computer system, the method
2 comprising the steps of:
3 storing, in a storage space, undo information for removing changes that are being
4 made by a plurality of entities, wherein the undo information for each entity of
5 the plurality of entities is stored in a segment of a plurality of segments within
6 said storage space, and a first entity of the plurality of entities is associated
7 with a first segment of the plurality of segments, and a set of one or more
8 entities of the plurality of entities is alone associated with a second segment of
9 the plurality of segments;
10 monitoring usage of the storage space by the entities in each period of time for a
11 series of periods of time; and
12 automatically adjusting at least one of the number of segments in the plurality of
13 segments and the sizes of the plurality of segments based on the usage, said
14 step of automatically adjusting comprising:
15 determining whether usage has decreased over a predetermined time based at
16 least in part on the usage in one or more periods of time of the series of
17 periods of time;
18 if usage has decreased over the predetermined time, then shrinking a sum of
19 the sizes of the plurality of segments;
20 determining whether sufficient storage space is already allocated to the first
21 segment for storing undo information included in a request from the
22 first entity; and

23 if it is determined that sufficient storage space is not already allocated to the
24 first segment, then increasing the size of the first segment by allocating
25 an additional amount of the storage space to the first segment, said
26 step of allocating the additional amount comprising,
27 determining whether the additional amount of the storage
28 space is available in storage space not currently
29 allocated to the plurality of segments, and if the
30 additional amount of the storage space is available in
31 storage space not currently allocated to the plurality of
32 segments, then obtaining the additional amount of
33 storage space from the storage space not currently
34 allocated; and
35 if the additional amount of the storage space is not available in
36 storage space not currently allocated to the plurality of
37 segments, then determining whether the additional
38 amount of the storage space is currently allocated to the
39 second segment of the plurality of segments and is not
40 used by the set of one or more entities, and if the
41 additional amount of the storage space is currently
42 allocated to the second segment and is not used by the
43 set, then obtaining the additional amount of storage
44 space by de-allocating from the second segment the
45 storage space currently allocated to the second segment
46 and not used by the set.

1 Claims 28-53 (Canceled).

1 54. (Currently Amended) A machine implemented method comprising the steps of:
2 storing undo information for removing changes that are being made to data in a
3 storage area that includes a plurality of segments, wherein application of the
4 undo information returns the data to which the changes were made to a state
5 of the data that existed prior to the changes;
6 using each segment of the plurality of segments as a circular buffer; and
7 automatically adjusting a size of the storage area based on usage of the plurality of
8 segments.

1 55. (Previously Presented) The method of Claim 54 wherein the step of automatically
2 adjusting is performed by at least adding a segment to the plurality of segments.

1 56. (Previously Presented) The method of Claim 54, wherein the step of automatically
2 adjusting is performed by at least increasing a size of at least one of the plurality of segments.

1 57. (Previously Presented) The method of Claim 54, wherein the step of automatically
2 adjusting includes at least de-allocating storage space in one segment of the plurality of
3 segments, wherein the one segment includes other storage space that is being used by one or
4 more entities.

1 58. (Previously Presented) The method of Claim 57, further comprising the step of
2 determining that the storage space is no longer being used by the one or the plurality of
3 segments; wherein the step of de-allocating is performed in response to the step of
4 determining.

1 59. (Previously Presented) The method of Claim 57, wherein the step of automatically
2 adjusting further includes at least allocating the storage space to another of the plurality of
3 segments.

1 60. (Previously Presented) The method of Claim 54, further comprising assigning one or
2 more entities to a segment based on the usage of the plurality of segments.

1 61. (Previously Presented) The method of Claim 54 further comprising collecting
2 statistical information to determine the usage of the plurality of segments.

1 62. (Previously Presented) The method of Claim 54, further comprising storing
2 information associated with the usage of the plurality of segments.

1 63. (Previously Presented) The method of Claim 62, wherein the information associated
2 with the usage is stored in an array.

1 64. (Previously Presented) The method of Claim 62, wherein the storing of the
2 information associated with the usage is performed periodically.

1 65. (Previously Presented) The method of Claim 54, wherein the plurality of segments
2 includes more than two segments, and the method further comprises the step determining the
3 usage of each the plurality of segments; and assigning an entity to one of the plurality of
4 segments based on the step of determining.

1 66. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 2.

1 67. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 3.

1 68. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 4.

1 69. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 5.

1 70. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 6.

1 71. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 7.

1 72. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 8.

1 73. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 10.

1 74. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 11.

1 75. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 12.

1 76. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 13.

1 77. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 14.

1 78. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 15.

1 79. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 16.

1 80. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 17.

1 81. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 18.

1 82. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 19.

1 83. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 20.

1 84. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 21.

1 85. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 22.

1 86. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 23.

1 87. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 24.

1 88. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 25.

1 89. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 26.

1 90. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 27.

1 91. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 54.

1 92. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 55.

1 93. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 56.

1 94. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 57.

1 95. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 58.

1 96. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 59.

1 97. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 60.

1 98. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 61.

1 99. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 62.

1 100. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 63.

1 101. (Previously Presented) A computer-readable medium carrying one or more sequences
2 of instructions which, when executed by one or more processors, causes the one or more
3 processors to perform the method recited in Claim 64.

- 1 102. (Previously Presented) A computer-readable medium carrying one or more sequences
- 2 of instructions which, when executed by one or more processors, causes the one or more
- 3 processors to perform the method recited in Claim 65.